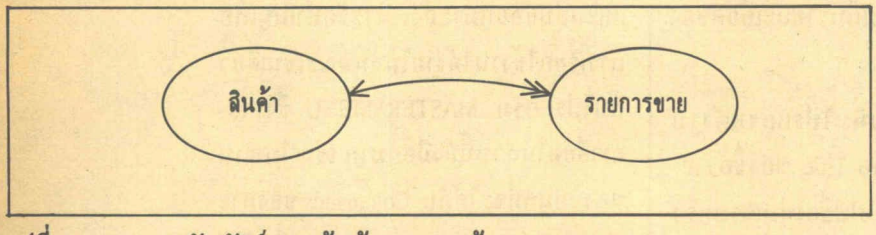
ดีเบสโฟร์ : การสร้างแฟ้มข้อมูลแบบรีเลชัน : โครงสร้างดีเบสโฟร์ และ SQL

ยืน ภู่วรวรรณ

การใช้งานระบบจัดการฐานข้อมูลมักเกี่ยวข้องกับแฟ้มหลาย ๆ แฟ้ม และที่สำคัญคือแฟ้มข้อมูลเหล่านั้นจะต้องมีความสัมพันธ์กัน โครงสร้างความสัมพันธ์กันนี้จะต้องง่ายต่อการใช้งาน ดีเบสโฟร์ได้รับการปรับปรุงจากเดิมจนสามารถจัดการกับแฟ้มข้อมูล ในลักษณะรีเลชันและใช้งานได้ง่ายขึ้น ลองดูจากตัวอย่างเช่น การบันทึกการขาย ซึ่งประกอบด้วยแฟ้มข้อมูลสินค้ากับแฟ้มข้อมูลลงรายการขาย รูปที่ 1 เป็นโครงสร้างของแฟ้มข้อมูลทั้งสองแฟ้มที่สัมพันธ์กัน

แฟ้มข้อมูลสินค้าเสมือนเป็นมาสเตอร์ไฟล์ ส่วนแฟ้มข้อมูลการขายทำหน้าที่เป็นทรานแซกชั่นไฟล์ ข้อมูลในแฟ้มสินค้าเก็บรายละเอียดของสินค้าแต่ละรายการไว้ ส่วนแฟ้มข้อมูลรายการขายเก็บข้อมูลบันทึกรายละเอียดการขาย หากมีการขายเกิดขึ้น 1 รายการก็จะบันทึกไว้ 1 เรคอร์ด ดังนั้นความสัมพันธ์ระหว่างสินค้ากับรายการขายจึงเป็นแบบ one to many โดยมีโครงร่างของแฟ้มข้อมูลดังรูปที่ 2

โครงสร้างของแฟ้มข้อมูลทั่วไปมีการสัมพันธ์กันแบบ one to many หรือแบบ many to many แต่สำหรับโมเดลที่เป็นรีเลชันแนลดาต้าเบส หากมีการเชื่อมแบบ many to many เราจะสร้างแฟ้มเชื่อมโยงให้เป็น one to many เพราะโครงสร้างการทำงานของระบบจัดการฐานข้อมูลเช่นดีเบสจะสนับสนุนการทำงานแบบ one to many ได้โดยตรง



รูปที่ 1 แสดงความสัมพันธ์ของแฟ้มข้อมูลสองแฟ้ม

แฟ้มสินค้า

รหัสสินค้า	ชื่อสินค้า	ราคาต่อหน่วย	ผู้รับผิดชอบ	Comment
ASG	DBASE	5,000.00	NHK	Watch closely
BPC	LOTUS	3,500.00	RBG	
FUR	WORD	2,000.00	ADN	
TAT	THAI	1,500.00	LPC	
UAB	CHART	3,000.00	NDR	

แฟ้มการขาย

รหัสสินค้า	วันที่ขาย	จำนวน	ผู้ขาย
ASG	02/11/88	5	YMN
ASG	05/11/88	2	TWN
ASG	09/11/88	3	YMN
ASG	04/12/88	4	KIK
TAT	28/10/88	5	SIS
TAT	28/10/88	2	TWN
TAT	12/12/88	4	TWN

รูปที่ 2 แสดงการเชื่อมโยงของแฟ้มข้อมูลในลักษณะ one to many

ดีเบสโฟร์กำหนดอินเด็กซ์ได้ในคำสั่ง CREATE

ในการสร้างฐานข้อมูลผู้ออกแบบระบบฐานข้อมูลต้องรู้โครงสร้างของฐานข้อมูลว่าประกอบด้วยแฟ้มข้อมูลอะไรบ้าง และแต่ละแฟ้มสัมพันธ์กันด้วยฟิลด์อะไร ซึ่งจะต้องสร้างอินเด็กซ์ฟิลด์สำหรับเชื่อมโยงระหว่างกันได้

การกำหนดอินเด็กซ์ในแฟ้มข้อมูลขณะสร้างนี้ หากกำหนดไว้หลายฟิลด์ดีเบสโฟร์จะสร้างใน .mdx ไฟล์ .mdx เป็นแฟ้มรวมที่เก็บของ .mdx ทำให้การอินเด็กซ์ทำได้หลายฟิลด์โดยไม่เปลืองพื้นที่ที่ใช้

การเรียกแฟ้มข้อมูลนี้ดีเบสโฟร์มีการเรียกอินเด็กซ์ที่แตกต่างจากของเดิม ได้พิจารณาการเปิดแฟ้ม product ที่มีฟิลด์ idproduct เป็นอินเด็กซ์

```
USE product ORDER idproduct
Master index: IDPRODUCT
```

อย่างไรก็ตาม ดีเบสโฟร์ยังคงสร้างพื้นที่สำหรับการเก็บแฟ้มข้อมูลในขณะเปิดใช้งานอยู่ 10 ที่อย่างเดิม แต่การใช้ .mdx ทำให้ประหยัดพื้นที่เก็บแฟ้มข้อมูลลงไปได้ พื้นที่แต่ละพื้นที่จะมีการกำหนดเป็น alias เหมือนดีเบสทรีพลัส แต่ดีเบสโฟร์

ให้ฟังก์ชันการตรวจสอบ alias ที่กำลังใช้อยู่ เช่น

```
. ? alias()
PRODUCT
```

การแก้ไขข้อมูลในดีเบสมีคำสั่งที่ได้ปรับปรุงไปจากเดิม เช่น Edit เราสามารถสั่งการแก้ไขข้อมูลโดยใช้วิธีการเลือกเรคอร์ดได้ เช่น

```
. EDIT 2
```

หมายถึงการเลือกเรคอร์ดที่ 2 มาแก้ไข ข้อมูลในเรคอร์ดที่ 2 จะปรากฏบนจอ โดยส่วนบนเป็นพูลดาวน์เมนูเพิ่ม คือ เราเลือกในพูลดาวน์เมนูได้ โดยกด ALT-R หมายถึงเลือก Record กด ALT-G หมายถึงเรียก GOTO เป็นต้น

คำสั่ง Edit นั้นจะปรากฏข้อความบนจอภาพดังนี้

```
  Records      Go To      Exit
IDPRODUCT    BPC
PNAME        LOTUS
PRICE_UNIT      3500.00
PER_INCHA    RBG
COMMENT      Watch closely
```

หากต้องการเลือกเฉพาะบางเรคอร์ดหรือบางฟิลด์มาแก้ไขก็ทำได้ เช่น หากต้องการได้ 3 เรคอร์ดแรก และเฉพาะฟิลด์ PNAME ก็ใช้คำสั่ง

```
. go top
PRODUCT: Record No       1
. EDIT next 3 field pname
```

การ edit นี้หากกด F2 จะกลับไป browse และเมื่ออยู่ที่ browse เรากด F2 ใหม่ก็กลับมาที่ edit ใหม่ สลับกันไปสลับกันมา

สำหรับคำสั่ง browse ก็ได้เพิ่มการแก้ไขให้ดียิ่งขึ้น เช่น ถ้าเราต้องการเรียกมาแก้ไขก็ใช้คำสั่ง

```
. browse
```

ตารางข้อมูลที่ปรากฏจะเป็นตารางที่สวยงามกว่าแบบเดิม อย่างไรก็ตาม หากเราต้องการจะล็อคบางฟิลด์ไว้กับจอภาพก็กระทำได้ โดยใช้คำสั่ง

```
. browse lock 2 .
```

หมายถึง การเลือก 2 ฟิลด์แรกไว้ ดังนั้นฟิลด์ Idproduct และ pname จะเป็นคอลัมน์ที่อยู่กับที่ ดังนั้นเมื่อเราเลื่อนเคอร์เซอร์ไปฟิลด์ทางขวาจนถึงฟิลด์สุดท้ายจะเห็นว่าฟิลด์ที่ 3 หายไป

นอกจากนี้ยังสามารถกำหนดให้มีการแก้ไขได้เฉพาะบางฟิลด์เท่านั้นด้วยการใช้ freeze เช่น

```
. browse freeze pname
```

หมายถึงการแก้ไขเฉพาะฟิลด์ pname เท่านั้น ฟิลด์อื่นใดนอกจากนี้จะไม่มีการแก้ไข

ผลของการ browse จะได้ผลดังนี้

แฟ้มสินค้า

รหัสสินค้า	ชื่อสินค้า	ราคาต่อหน่วย	ผู้รับผิดชอบ	Comment
ASG	DBASE	5,000.00	NHK	Watch closely
BPC	LOTUS	3,500.00	RBG	
FUR	WORD	2,000.00	ADN	
TAT	THAI	1,500.00	LPC	
UAB	CHART	3,000.00	NDR	

การใช้ SELECT

การเลือกแฟ้มข้อมูลในพื้นที่อื่นยังคงเหมือนดีเบสทรีพลัสคือ ใช้คำสั่ง SELECT เช่น ใช้

```
. SELECT 2
. USE SALE
```

ขณะนี้ดีเบสอยู่ในพื้นที่ 2 ดังนั้นเมื่อใช้คำสั่งการทำงานเกี่ยวกับแฟ้มข้อมูลก็จะปรากฏอยู่ในแฟ้ม sale เช่น

```
. DISPLAY STRUCTURE
Structure for database: C:\DB4\SALE.DBF
Number of data records:      7
Date of last update    : 22/01/89
Field  Field Name  Type       Width    Dec  Index
    1  IDPRODUCT   Character      3           Y
    2  SDATE       Date           8           N
    3  AMT         Numeric        5           N
    4  SPERSON     Character      3           N
** Total **                     20

. LIST
Record#  IDPRODUCT SDATE      AMT SPERSON
      1  ASG       02/11/88     5 YMN
      2  ASG       05/11/88     2 TWN
      3  ASG       09/11/88     3 YMN
      4  ASG       04/12/88     4 KIK
      5  TAT       28/10/88     5 SIS
      6  TAT       28/10/88     2 TWN
      7  TAT       12/12/88     4 TWN
```

การเลือก Alias

การเลือกพื้นที่นั้นจะใช้ SELECT แล้วตามด้วยหมายเลขก็ได้ แต่ก็สามารถใช้ชื่อย่อของ alias นี้ได้เช่นกัน เช่น

```
. SELECT A
. DISPLAY STRUCTURE
Structure for database: C:\DB4\PRODUCT.DBF
Number of data records:      5
Date of last update    : 22/01/89
Field  Field Name  Type       Width    Dec  Index
    1  IDPRODUCT   Character      3           Y
    2  PNAME       Character     30           N
    3  PRICE_UNIT  Numeric       10     2     N
    4  PER_INCHA   Character      3           N
    5  COMMENT     Character     30           N
** Total **                     77
```

นอกจากนี้ดีเบสโฟร์ยังรวมคำสั่งให้เราใช้จาก 2 บรรทัดเหลือเพียงบรรทัดเดียว คือ การใช้คำสั่ง use in เช่น

```
. USE SALE IN 2
```

หมายถึง select 2 แล้ว use sale นั่นเอง และการตั้งชื่อ alias นั้นยังมาใช้ได้กับคำสั่ง use in ด้วย เช่น

```
. USE SALE IN 2 ALIAS SREG
```

การสร้างความสัมพันธ์ระหว่างแฟ้ม

ในดีเบสทรีพลัสมีคำสั่ง set relation แล้ว แต่สำหรับดีเบสโฟร์การเซตรีเลชันทำได้ง่ายขึ้น และเชื่อมโยงไปกับคำสั่ง LIST หรือ display ได้ และนำไปใช้กับคำสั่ง View หรือ Query ได้เช่นกัน ลองดูตัวอย่าง

```
. SELECT sreg
. SET ORDER TO TAG idproduct
Master index: IDPRODUCT
. SELECT 1
. USE product
. SET ORDER TO TAG idproduct
Master index: IDPRODUCT
. SET RELATION TO idproduct INTO sreg
```

ในกรณีนี้แฟ้มข้อมูลทั้งสองจะสัมพันธ์กันตามรูปแบบที่ต้องการ และหากเราต้องการให้มีการแสดงทางด้านแฟ้มการขายหลาย ๆ เรคอร์ดที่เชื่อมกับแฟ้มสินค้า ทำได้ดังนี้คือ

```
. SET SKIP to sreg
. LIST idproduct,pname,sreg->sdate,sreg->amt
```

ผลลัพธ์ที่ได้จะเป็นดังนี้

```
Record#  idproduct pname    sreg->sdate sreg->amt
      1  ASG       DBASE    02/11/88            5
      1  ASG       DBASE    05/11/88            2
      1  ASG       DBASE    09/11/88            3
      1  ASG       DBASE    04/12/88            4
      2  BPC       LOTUS      /  /
      3  FUR       WORD       /  /
      4  TAT       THAI     28/10/88            5
      4  TAT       THAI     28/10/88            2
      4  TAT       THAI     12/12/88            4
      5  UAB       CHART      /  /
```

การสร้างโมเดลฐานข้อมูลนี้จำเป็นต้องเชื่อมแฟ้มข้อมูลหลาย ๆ แฟ้มเข้าด้วยกัน เพื่อให้เกิดโครงร่างเป็นฐานข้อมูลที่แท้จริง และสามารถเรียกค้นข้อมูลตามความสัมพันธ์นั้นได้ ดีเบสโฟร์ได้เน้นจุดนี้พอควรโดยให้ผู้สร้างแฟ้มสามารถกำหนดอินเด็กซ์ได้ตั้งแต่เริ่มแรก และเรียกข้อมูลที่สัมพันธ์กันได้เช่นกัน

โครงสร้างดีเบสโฟร์และ SQL

ดีเบสโฟร์มีโครงสร้างการพัฒนาที่ก้าวล้ำไปจากดีเบสทรีพลัสมาก ดีเบสโฟร์มีโมดูลการจัดการฐานข้อมูลที่เป็นแบบรีเลชันแนลได้สมบูรณ์แบบ โครงสร้างของดีเบสโฟร์แบ่งออกเป็น 3 ส่วนใหญ่ ๆ คือ

คำสั่งโหมด (dot prompt) ดีเบสโฟร์มีชุดคำสั่งเหมือนกับดีเบสทรีโดยเพิ่มจำนวนคำสั่งและฟังก์ชันที่ใช้ให้มากขึ้นและสะดวกในการใช้งานได้ดี การจัดการข้อมูลในกรณีนี้จะจัดการข้อมูลในไฟล์ที่มีนามสกุลเป็น .dbf ส่วนการสร้างดัชนีจะใช้ไฟล์ชื่อ .ndx หรือ .mdx ในการจัดการด้วยคำสั่งนี้สามารถนำคำสั่งมารวมกันไว้เป็นโปรแกรมที่เก็บในไฟล์ที่มีนามสกุล .prg โครงสร้างของระบบฐานข้อมูลที่จัดการเป็นแบบรีเลชันแนลโดยจัดการข้อมูลในระดับเรคอร์ด และเชื่อมโยงกันได้ด้วยคำสั่ง set relation โครงสร้างของระบบฐานข้อมูลเป็นดังรูปที่ 1

คำสั่งการจัดการข้อมูลในศูนย์กลางการควบคุม (Control Center) ดีเบสโฟร์ได้พัฒนารูปแบบการเชื่อมต่อกับผู้ใช้ไว้อย่างดีมาก โดยสร้างระบบ Control Center ขึ้นมาใหม่ คอนโทรลเซนเตอร์หรือศูนย์กลางการควบคุมนี้เป็นโปรแกรมที่ทางบริษัทแอสดันเทตเขียนไว้ โดยมีโครงสร้างการจัดการหลายเรื่องคือ

ส่วนจัดการข้อมูล เป็นส่วนที่ทำการสร้างไฟล์การแก้ไขข้อมูล การเพิ่ม/ลบข้อมูล

ส่วนการ Queries เป็นส่วนให้ผู้ใช้สร้างวิวสำหรับการเรียกค้นข้อมูล

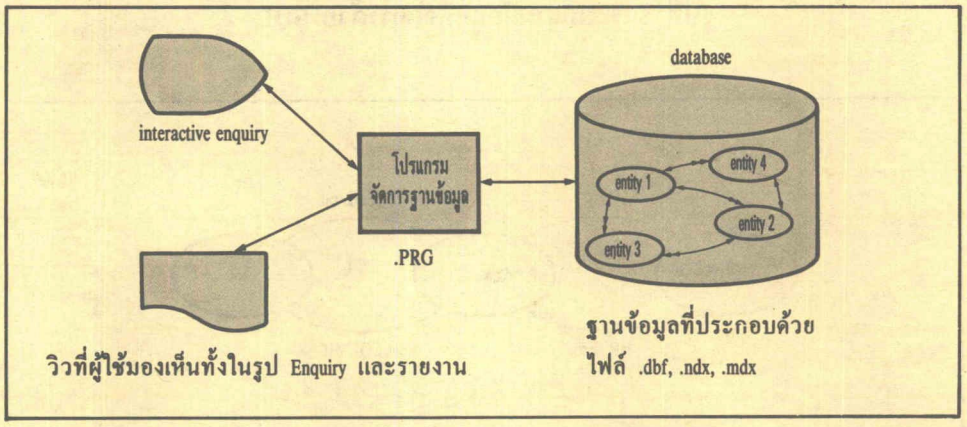
ส่วนของ Form เป็นส่วนการสร้างรูปแบบของจอภาพหรือรายงานตามที่ต้องการ

ส่วนของ Report คือโมดูลที่นำข้อมูลจากแฟ้มต่าง ๆ มาสร้างรายงาน

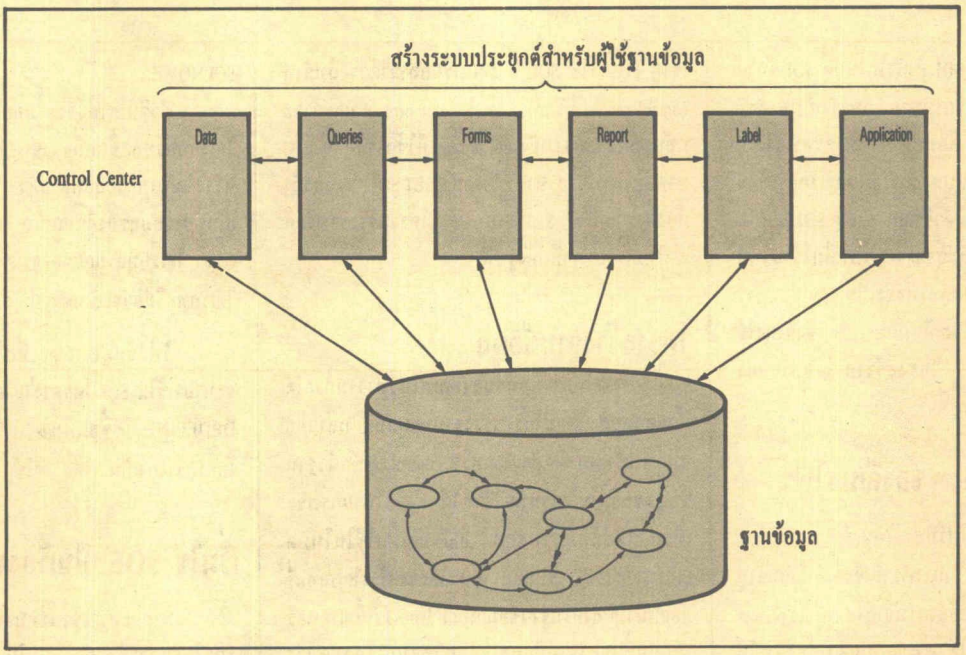
ส่วนของ Label เป็นส่วนที่ใช้ในการสร้างรูปฟอร์มสำหรับพิมพ์จ่าหน้าซอง

ส่วน Application เป็นส่วนสร้างเมนูให้ผู้ใช้ใช้งานได้ตรงกับความต้องการเหมือนการเขียนโปรแกรม

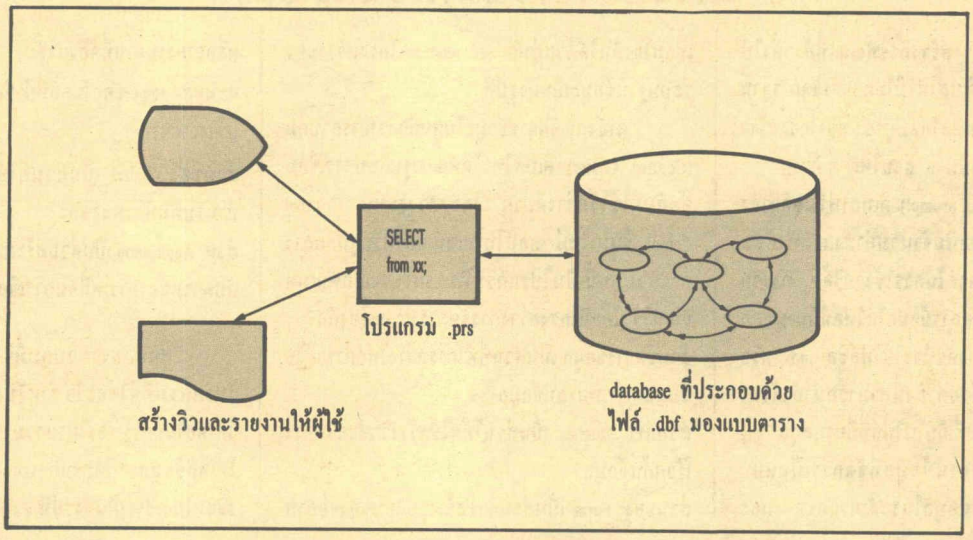
ศูนย์กลางควบคุมนี้จึงเปรียบเสมือนระบบโปรแกรมที่ผู้ใช้จะใช้สำหรับการจัดการฐานข้อมูลได้โดยไม่ต้องรู้เรื่องโปรแกรม แต่ต้องรู้เรื่องการสร้างโมเดลข้อมูล ไดอะแกรมของระบบฐานข้อมูลที่ใช้เขียนโมเดลจะเป็นดังรูปที่ 2



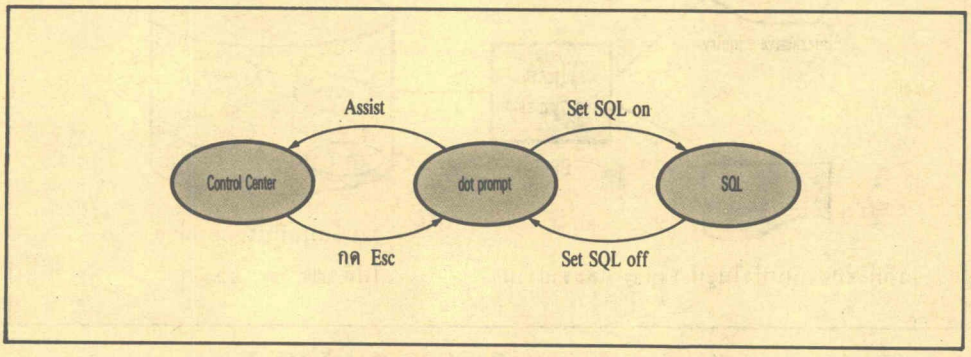
รูปที่ 1 โมเดลการจัดการฐานข้อมูลแบบรีเลชันของดีเบสโฟร์ในโหมด dot prompt



รูปที่ 2 การจัดการฐานข้อมูลด้วย Control Center



รูปที่ 3 ระบบฐานข้อมูลที่จัดการด้วย SQL



รูปที่ 4 ไดอะแกรมการเลือกโหมด

คำสั่งในโหมด SQL คำสั่งในโหมด SQL จัดได้ว่าเป็นคำสั่งที่มีประสิทธิภาพมาก เพราะเป็นคำสั่งที่ทำงานในระดับเซต คือกลุ่มของเรคอร์ดหลาย ๆ เรคอร์ด ในคำสั่ง dot|ดีเบสจะกระทำทีละเรคอร์ดโดยมีเรคอร์ดพอยน์เตอร์เป็นตัวชี้บอก คำสั่ง SQL จะเป็นคำสั่งที่ใช้งานได้ดี และสามารถนำมาเก็บไว้ในไฟล์เป็นโปรแกรม โดยนามสกุลไฟล์จะเป็น .prs ส่วนการจัดการข้อมูลซึ่ง SQL จะเห็นเป็น table ซึ่งยังคงใช้นามสกุล .dbf ดังเดิม โครงสร้างการจัดการของ SQL จะเป็นดังรูปที่ 3

การเข้าสู่โหมดต่าง ๆ ของดีเบสโฟร์

เนื่องจากดีเบสโฟร์มีสามโหมดคือ dot prompt, control center และ SQL ในการเริ่มต้นจึงต้องเลือกเข้าสู่โหมดใดโหมดหนึ่ง หากต้องการเข้าสู่โหมด dot prompt ใน config.db ก็ไม่ต้องใส่ command แต่ถ้าต้องการให้เข้าสู่ control center ก็จะใช้ command = assist ถ้าต้องการเข้าสู่ SQL ก็ใช้ SQL = ON หรือกล่าวได้ว่า ถ้าเข้าสู่ control center ก็เรียก assist ใน dot prompt และถ้าจะกลับจาก control มายัง dot.prompt ก็ใช้การกดคีย์ ESC หากต้องการเข้าสู่ SQL ก็ใช้คำสั่ง SET SQL ON และถ้าต้องการเลิกก็ใช้ SET SQL OFF โครงสร้างการเรียกเข้าโหมดต่าง ๆ เป็นดังรูปที่ 4

โครงสร้างฐานข้อมูล

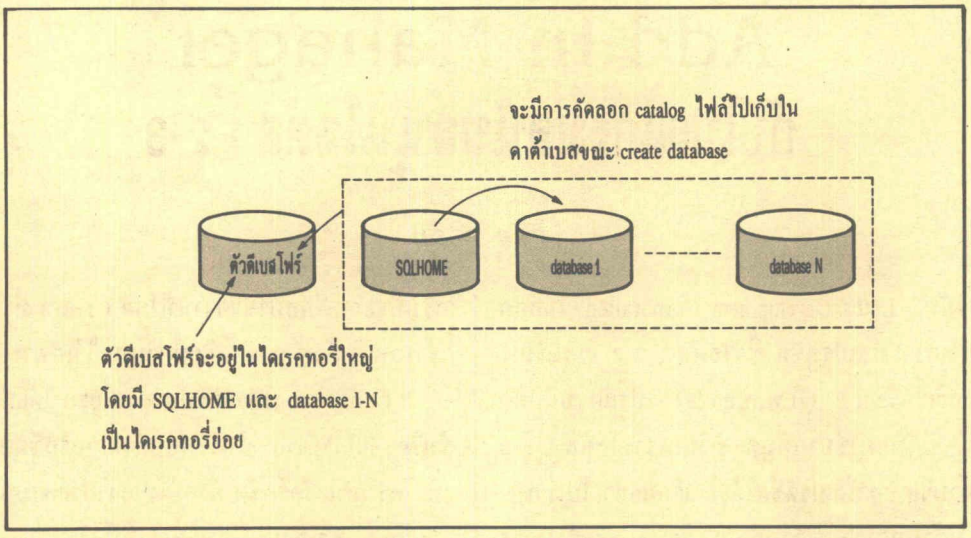
การจัดการข้อมูลของดีเบสโฟร์ทำงานภายใต้เอ็มเอสดอส โดยมีโครงสร้างฐานข้อมูลที่ดี กล่าวคือ เมื่อใช้ในโหมด dot prompt หรือ control center ตัวฐานข้อมูลจะอยู่ในไดเรคทอรี่ใดก็ได้ โดยโปรแกรมจะเห็นเป็นไฟล์ที่มีนามสกุล .dbf แต่เมื่อใช้ในโหมด SQL การสร้างฐานข้อมูล ดีเบสโฟร์จะสร้างไดเรคทอรี่ย่อยหนึ่ง ๆ สำหรับฐานข้อมูลหนึ่ง โดยตั้งชื่อไดเรคทอรี่นั้นเป็นชื่อของฐานข้อมูล โดยมีส่วนการควบคุมว่าขณะนี้มีกี่ฐานข้อมูล และจะจัดการส่วนนี้ไว้ที่ไดเรคทอรี่ SQLHOME

ตัวดีเบสโฟร์จะเก็บอยู่ในฮาร์ดดิสก์โดยใช้ชื่อไดเรคทอรี่หลักชื่อหนึ่ง เช่น dbase4 ตัวดีเบสโฟร์จะสร้างไดเรคทอรี่ย่อยหลายตัว เช่น ไดเรคทอรี่ย่อยเก็บฐานข้อมูลของดีเบสแบบ dot prompt หรือ control center ไดเรคทอรี่ย่อยของฐานข้อมูล SQL ไดอะแกรมไดเรคทอรี่ย่อยจะเป็นดังรูปที่ 5

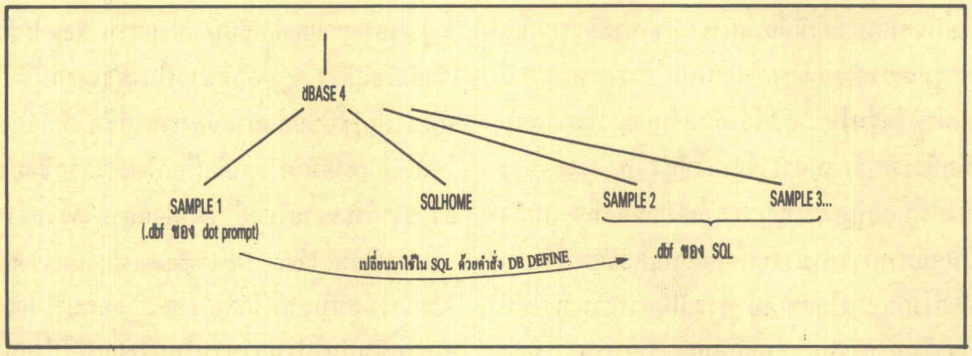
ไฟล์ข้อมูล .dbf ที่สร้างโดย .dot prompt จะนำมาใช้ใน SQL โดยตรงไม่ได้ ต้องเปลี่ยนด้วยคำสั่ง DBDEFINE <ชื่อแฟ้ม> จะทำให้เกิดการเปลี่ยนโครงสร้างมาเป็น .dbf ที่ใช้กับ SQL ได้ดังรูปที่ 6

เริ่มใช้ SQL เป็นเรื่องง่าย

หากจะสร้างฐานข้อมูลที่ใช้กับ SQL เป็นเรื่องง่าย โดยเราจะต้องออกแบบโมเดลของฐานข้อมูลก่อน ขั้นตอนของการปฏิบัติงานเริ่มต้นดังนี้



รูปที่ 5 โครงสร้างการจัดฐานข้อมูล



รูปที่ 6

1. การติดตั้งดีเบสโฟร์จะต้องติดตั้งโมดูล SQL ลงไปด้วย โดยโมดูลของ SQL จะอยู่ที่ SQLHOME ส่วนนี้จะเป็นการสร้าง SQL.SYS หรือโปรแกรมระบบหลายตัวไว้ในไดเรคทอรี่ย่อยนี้

2. เมื่ออยู่ที่ .dot prompt ใช้คำสั่ง

.SET SQL ON

ดีเบสจะเปลี่ยนเครื่องหมาย prompt เป็น SQL.

3. หากยังไม่มีฐานข้อมูลที่สร้างขึ้นโดย SQL เลย หากต้องการสร้างขึ้น เช่น อยากสร้างฐานข้อมูล INVENT ก็ทำได้โดยใช้คำสั่ง

SQL.CREATE DATABASE INVENT;

ดีเบสโฟร์จะสร้างไดเรคทอรี่ย่อย INVENT สำหรับเก็บฐานข้อมูล INVENT โดยมีการคัดลอกแฟ้มบางส่วนจาก SQLHOME มาเก็บไว้ ส่วนนี้เรียกว่าส่วนกำหนด data dictionary

4. เราจะเริ่มแสดงดูว่ามีฐานข้อมูลอะไรด้วยการใช้คำสั่ง

SQL.SHOW DATABASE;

ดีเบสจะแสดงชื่อฐานข้อมูล (ชื่อฐานข้อมูลไม่ใช่ชื่อไฟล์ แต่เป็นชื่อไดเรคทอรี่ย่อย)

5. หากจะใช้ฐานข้อมูลตัวใดก็ใช้คำสั่งเปิดฐานข้อมูล เช่น ต้องการใช้ INVENT

/SQL.START DATABASE INVENT;

SQL จะให้ชื่อไว้ใน status line ตอนล่างว่าใช้ฐานข้อมูลอะไร

6. หากจะสร้างตารางข้อมูลลงในฐานข้อมูล เราใช้คำสั่ง create table เช่น ต้องการสร้างตาราง product มี 2 ฟิลด์

SQL.CREATE TABLE product (ID char (4), name char (20));

SQL จะสร้างตารางให้

7. เมื่อจะป้อนข้อมูลโดยใช้คำสั่ง INSERT INTO สมมติป้อนข้อมูล ID = A423 และชื่อสินค้า STRAWBERRY จะใช้คำสั่ง

SQL.INSERT INTO product (id, name) VALUES ("A423", "STRAWBERRY");

8. เมื่อต้องการเรียกใช้ด้วยคำสั่ง SELECT ลองดู

SQL. SELECT * FROM product;

อนึ่ง คำสั่ง SQL ทุกตัวจะต้องปิดท้ายด้วย ; เสมอ การใช้ SQL. ในดีเบสโฟร์ไม่ใช่เรื่องยาก ผู้อ่านลองดูสิครับแล้วจะพบว่า ดีเบสโฟร์จัดการฐานข้อมูลได้ดีเพียงไร และที่สำคัญคือเราสามารถใช้คำสั่งใน .dot prompt มาผสมกับ SQL ได้ ทำให้การทำงานดีเยี่ยมขึ้นอีกมาก เช่น ใช้ IF ใช้ DOWHILE ในโปรแกรมของ SQL ได้อีกด้วย

โปรดติดตามในวารสารไมโครฯต่อไป เราจะนำเอาเทคนิคการใช้ SQL ในการจัดการฐานข้อมูลแบบรีเลชันแนลมาเสนอต่อไป